

REMARKS**The Present Invention**

The present invention is a method and apparatus for updating a shared session database that is accessible by multiple servers. For instance, Web sites often divide the tasks of servicing requests into a three tier system with a different server or plurality of servers to handle each tier. Since http is a connectionless protocol, one request from a particular client can be directed to one application server while the next request from the same client machine might be directed to a different application server. Accordingly, a means must be provided for the various servers to access the session data developed by another server. Commonly, such sharing of http session data is enabled by use of a database server that is accessible to the plurality of application servers for storing session data. Particularly, an application server stores session data in local memory, but also writes a copy of the session data to the session database. If a different server services a request from a client, that different server can go to the database and read the session data for the corresponding session. The session data is updated in both the local memory and the database each time a request causes a change in the data. The writing of such session data to the shared database can consume a large amount of bandwidth on the network.

The present invention reduces the number of writes to an http session database in order to conserve system resources. Specifically, while each server continues to update the http session data in its local memory every time there is a change in the session data, it writes a copy of the session data to the shared database only at designated times. In one embodiment, the designated time is periodic. In another, it is

after a specified number of requests in that session have been received. In yet another it is after a specified number of changes to the session data have been made.

The Final Office Action

In the final Office Action dated December 15, 2006, the Office has maintained all rejections asserted in the previous Office Action. Accordingly, all claims, claims 1-24, remain rejected with ball of the rejections relying on at least a proposed combination of the Courts reference and the Dharmarajan reference.

Specifically, the Office rejected claims 11-13, 16, and 17 as obvious over Courts in view of newly cited U.S. Patent No. 7,010,605 issued to Dharmarajan (hereinafter Dharmarajan); claim 14 as obvious over Courts, Dharmarajan and Prabandham; and claim 1-10, 15, 18, 22, and 23 as obvious over Courts, Dharmarajan, Prabandham and Ng.

Response to Rejections**Background**

In response to the previous Office Action, Applicant noted that these rejections were very similar to the previous rejections, except for the added citation of the Dharmarajan reference. Particularly, the Office had previously been relying on the Courts reference as teaching that the writing of the session data to the shared database was performed "at a designated time that is a function of a predetermined time interval since her last right to send database..." (as recited in the independent claims 1 and 11) or was "a function of at least one of (a) the number of times the httpsession object data

is updated in said local memory and (b) the number of times an http request in said http session is serviced" (as recited in independent claim 18). More particularly, the Office eventually conceded that Courts, in fact, did not teach this feature, but added the Dharmarajan reference and asserted that it teaches this feature and that it was obvious to combine the two references.

In response to the previous Office section, Applicant argued that the Dharmarajan reference, in fact, also does not teach in the above-quoted features.

In reply, the Office has repeated the previous rejections and asserted that Applicant's arguments were unpersuasive because Applicant allegedly argued against the references individually, whereas it is the combination of Courts and Dharmarajan that teaches the above-quoted claim elements.

Applicant respectfully submits that its previous arguments (as well as the arguments to follow) do, in fact, address the proposed combination and not the references individually.

MPEP § 2143 lists the three requirements for a proper rejection based on the combination of two or more references. They are:

First, there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art to modify the reference or to combine the reference teachings. Second, there must be a reasonable expectation of success. Finally, the prior art reference (or references when combined) must teach or suggest all the claim limitations.

The issue that Applicant addressed in the previous response (and continues to address in this response) is whether Courts and Dharmarajan collectively teach writing the data to the shared database at intervals based on any one of (a) an elapsed time since the last write to the shared database, (b) the number of times the httpsession

object data is updated in the local memory, or (c) the number of times and http request in said http session is serviced.

The Courts Reference

As previously discussed and as eventually conceded to by the Office, Courts teaches nothing more than writing the session data to the global database every time the local session data is updated, exactly as described as the prior art in the Background of the Invention section of the present application.

The Dharmarajan Reference

Furthermore, while Dharmarajan generically teaches a timer being set to elapse after a predetermined amount of time, that timer has absolutely nothing to do with writing session data to a shared database.

Dharmarajan describes a technique for enhanced security during a session between a server and a client in which the server is using information obtained from a cookie received from the client. Specifically, Dharmarajan addresses a drawback of previous cookie schemes as described in col. 2, lines 39-53, quoted below:

Another drawback to using cookies stems from the fact that cookies are transmitted in the open from the client computer to the server computer. Because cookies are transmitted in the open over the Internet, there is a possibility that the cookies may be intercepted by an unauthorized recipient. An intercepted cookie may then be "replayed" by the unauthorized recipient to gain improper access to the Web server. Col. 2, lines 39-42.

In accordance with the relevant portions of Dharmarajan, the server starts a session timer (step 1002 of Fig. 10) and waits for it to run out (step 1004). When it runs out, it authenticates the cookie by requesting the cookie again from the client (step

1006-1008), decoding and decrypting the cookie again (step 1010), checking if the data is valid (step 1012). If so, it generates a new encrypted cookie and sends it to the client (step 1014) and keeps the session going. The process continuously repeats until the session is ended (see steps 1016 and 1004). However, if the data in the cookie is not valid (step 1012), the server ends the session (step 1018) assuming that the cookie (and therefore the session) is fraudulent.

Thus, Dharmarajan's timer counts the time that the server waits between authenticating the cookie (and, if authenticated generating a new cookie and sending it to the client).

Dharmarajan teaches writing encrypted cookies from a server to a client machine at predetermined time intervals. This has absolutely no relevance to writing session data from a local database to a shared database at predetermined intervals, as in the present invention.

The Proposed Combination of Courts and Dharmarajan

The issue at hand is whether the references, in combination, suggest replacing the prior art scheme of Courts (i.e., updating the global http session database every time the local http session database is updated) with a scheme in which the global http session database is updated after any one of (1) an elapsed time interval, (b) a certain number of http session object data updates since the last write to the shared database, or (c) servicing a certain number of http requests since the last write to the shared database.

A reference, such as Dharmarajan, that discloses sending a new encrypted cookie from a server to a client machine at predetermined time intervals in order to enhance security simply does not provide this teaching. Rather, it is utterly irrelevant. It is no more relevant than a reference that disclosed boiling an egg for three minutes, followed by boiling another egg for three minutes, followed by boiling a next egg for three minutes.

Simply put, sending a new encrypted cookie to a client machine at predetermined, timed intervals in order to prevent unauthorized interception and use of those cookies could not possibly lead one to the realization that one should update http session data in a shared database at intervals other than the conventional interval of every time the http session data is updated locally. The two subjects have no more to do with each other, than they do with boiling an egg every three minutes.

Thus, the proposed combination does not teach or suggest to one of ordinary skill in the related arts “writing a copy of said data for each said session stored in said local memory into a central memory accessible to all servers of said server system at designated times, said designated times being a function of a predetermined time interval since a last write to said database of data for said sessions” as recited in claim 11 or “a second computer program adapted to write to said database a copy of said HttpSession data for each said http session at a designated time that is a function of a predetermined time interval since a last write to said database of HttpSession object data for said http session”.

With respect to independent claim 18, the patentable distinctions are even more glaring. Specifically, claim 18 recites different criteria for the interval between updates

of the shared http session database than those recited in claims 1 and 11. Claim 18 recites that “said designated times [are] determined as a function of at least one of (a) the number of times the http session object data is updated in said local memory and (b) the number of times said http request in said http session is serviced”.

Dharmarajan’s timer waits a predetermined period of time before updating the cookie. Thus, with respect to claim 18, not only does Dharmarajan disclose absolutely nothing about using a timer to update a shared http session database, but it also discloses absolutely nothing relevant to the specific timing criteria recited in the claim, namely, either the number of times the http session object data is updated in said local memory or the number of times said http request in said http session is serviced. Rather, Dharmarajan discloses a predetermined elapsed time.

Thus, the independent claims distinguish over the prior art of record because none of the references taken alone or in combination suggest updating a shared http session database based on any of the timing criteria recited in the independent claims of the present application. The core concept of the present invention is simply lacking from the prior art of record.

In view of the foregoing amendments and remarks, this application is now in condition for allowance. Applicant respectfully requests the Examiner to issue a Notice of Allowance at the earliest possible date. The Examiner is invited to contact

PATENT

ATTORNEY DOCKET No. RSW9-2001-0063-US1
Patent Application No. 09/823,120

Applicant's undersigned counsel by telephone call in order to further the prosecution of this case in any way.

Respectfully submitted,

Dated: February 13, 2007

/Theodore Naccarella/
Theodore Naccarella
Registration No. 33,023
Synnestvedt & Lechner LLP
2600 Aramark Tower
1101 Market Street
Philadelphia, PA 19107
Telephone: (215) 923-4466
Facsimile: (215) 923-2189

TXN:pmf

S:\IBM\IBM RALEIGH RSW\PATENTS\IP24881 USA\PTO\RESPONSE TO 12-15-06 OFFICE ACTION.DOC